1

METHOD AND APPARATUS FOR GENERATING AND CONTROLLING
A QUADRATURE CLOCK

BACKGROUND

Digital logic circuits frequently rely on clock signals for
synchronization, derivation of reference signals, measuring phase differences,
and other functions. Some applications require a quadrature clock that has a
90° phase difference from a reference clock. A quadrature clock is frequently
used for strobe signals, for example.

A centralized quadrature clock may be generated and distributed to all
components. In particular, a first clock and a second clock having a 90° phase
difference from the first clock are centrally generated and distributed. One
disadvantage of this approach is that clock signals tend to have constraints
that are difficult to maintain throughout the distribution when the
distribution is over a relatively large area or used to drive a relatively large
number of components. For example, significant consumption of die area
and wire routing resources are required to meet skew requirements for both
clocks in integrated circuit applications. The use of some types of integrated
circuit logic in the clock distribution buffers may even result in a doubling of
the power consumption due to the distribution of two clocks.

Another technique for generating the quadrature clock entails
distributing a reference clock signal to regions of an integrated circuit. Each
region has a local phase locked loop (PLL) or local delay locked loop (DLL) to
derive the quadrature clock signal from the reference clock signal. A
disadvantage of this approach is that the localized PLLs or DLLs introduce
complexity into the integrated circuit design thus incurring more design and
verification time. In addition, co-ordinating selection from the reference clock
of the proper phase associated with the first and second clocks across all
regions such that the first clock in one region is in phase with the first clock of
another region is problematic.

SUMMARY

In view of limitations of known systems and methods, various methods and apparatus for generating and controlling a quadrature clock are described.

5   In one embodiment, a quadrature clock generating apparatus includes a multiplexer selecting one of a generated clock and a gated generated clock as a double clock in accordance with a halt multiplexer control. Divider circuitry provides an alignment signal corresponding to an inverted double clock divided by two. A recovery circuit recovers first and second clocks

10   having a 90° phase difference from the double clock in accordance with the alignment signal. A halt circuit controls the halt multiplexer control to select the gated generated clock when the alignment signal matches a pre-determined clock level. The halt multiplexer control is clocked by the generated clock.

15   One embodiment of a method of generating a quadrature clock includes selecting one of a generated clock and a gated generated clock as a double clock in accordance with a halt multiplexer control, wherein the double clock has a frequency twice that of a received reference clock. An alignment signal corresponding to an inverted double clock divided by two is

20   generated. A plurality of instances of a first clock and an associated second clock are derived from the double clock in accordance with the alignment signal. Each first and associated second clock instance have a 90° relative phase difference. The halt multiplexer control is controlled to select the gated generated clock when the alignment signal matches a pre-determined clock

25   level, wherein the halt multiplexer is clocked by the generated clock.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings, in which like references indicate similar elements and in which:

30   Figure 1 illustrates one embodiment of quadrature clock generation circuitry.

Figure 2 illustrates one embodiment of waveforms associated with the clock generation circuitry of Figure 1.

Figure 3 illustrates one embodiment of quadrature clock generation circuitry providing a multiple instances of a quadrature clock.

Figure 4 illustrates one embodiment of a method for generating a quadrature clock.

Figure 5 illustrates one embodiment of an apparatus for generating and controlling the quadrature clock.

Figure 6 illustrates one embodiment of a halt control circuit for controllably halting and stepping the quadrature clock.

Figure 7 illustrates one embodiment of a method for generating and controlling the quadrature clock.

Figure 8 illustrates one embodiment of quadrature clock generation and distribution across an integrated circuit die within an integrated circuit package.

DETAILED DESCRIPTION

Figure 1 illustrates one embodiment of quadrature clock generation circuitry for generating a first clock CLK1 180 and a second clock CLK2 190. The second clock has a quadrature relationship to the first clock. In one embodiment, the clock generation circuitry resides on an integrated circuit die 196.

Clock generator 110 generates an output 116 having a frequency that is a multiple, m, of the frequency of a reference clock, REFCLK 112. In the illustrated embodiment, clock generator 110 is a phase locked loop (PLL). The PLL has a feedback path including inverter 182, divider 170, and flip-flop 150.

As a result of the divide-by-m divider, the frequency of the PLL output 116 is m times the frequency of REFCLK 112. The output of divider 170 has a frequency that is $1/m$ that of the PLL output.

In the illustrated embodiment, $m=2$ such that the PLL output is twice the frequency of REFCLK. PLL 110 is thus a clock generator that provides a double clock (2XCLK 116) that has a frequency twice that of the received reference clock, REFCLK.

4

In one embodiment, this 2XCLK 116 is distributed to the flip-flops 150, 160 in a differential form. Thus single-sided-to-differential converter 120 converts the single sided 2XCLK to a differential 2XCLK signal. Proximate the flip-flops, the differential-to-single-sided converter 130 converts the

5   differential 2XCLK back to a single-sided 2XCLK.

Divider 170 and inverter 172 form divider circuitry for providing an alignment signal 1XALIGN 176 having a frequency that is $1/m$ that of the PLL output. In particular, the divider circuitry provides a 1XALIGN corresponding to an inverted PLL output divided by m. Given that the PLL

10   output has a frequency that is m times the REFCLK frequency, the signal 1XALIGN 176 has a frequency that is the same as that of the REFCLK. Regardless of the value of m, the 1XALIGN will always have a frequency that is the same as that of the REFCLK.

In the illustrated embodiment, $m=2$ such that 1XALIGN corresponds to

15   an inverted double clock divided by two. In the illustrated embodiment, the inverter 172 inverts the double clock such that the divider 170 receives an inverted double clock.

The 1XALIGN 176 is provided as a data input to flip-flop 150. The output of flip-flop 150 is the first clock CLK1 180. CLK1 is provided as a

20   feedback input to PLL 110.

The double clock (i.e., 2XCLK) is used to clock flip-flop 150. The double clock is inverted by inverter 140 and the inverted double clock (i.e., $\overline{\text{2XCLK}}$) is used to clock flip-flop 160. Flip-flop 150 provides the first clock signal, CLK1 180. Flip-flop 160 provides the second clock signal, CLK2 190. The

25   second clock has a 90° phase displacement from the first clock. In one embodiment, each of flip-flops 150, 160 is a D-type flip-flop.

PLL 110 drives the PLL output so as to minimize the phase difference between REFCLK and CLK1. Thus even if CLK1 and REFCLK are initially out of phase, PLL 110 will drive the PLL output to bring CLK1 and REFCLK

30   into phase alignment.

In one embodiment, the clock generator, divider circuitry and recovery circuitry reside on a same integrated circuit die. In one

embodiment, the clock generator, divider circuitry and recovery circuitry are formed as a metal oxide semiconductor field effect transistor (MOSFET) integrated circuit.

Figure 2 illustrates waveforms for various signals present in the

5   quadrature clock generation circuitry of Figure 1. The 2XCLK waveform (b) has twice the frequency of REFCLK illustrated as waveform (a). The 1XALIGN waveform (c) has half the frequency as the double clock 2XCLK waveform. Accordingly, 1XALIGN and REFCLK have the same frequency. As a result of the inverted 2XCLK presented to the divider, the leading edge

10   of the 1XALIGN signal is aligned with a falling edge of the 2XCLK signal.

Given that the CLK1 signal of waveform (d) is derived by clocking a flip-flop receiving the 1XALIGN signal, the CLK1 signal of waveform (d) will transition to the value of 1XALIGN on the rising edge of the 2XCLK.

The CLK2 signal will transition to the value of CLK1 on the rising edge

15   of $\overline{\text{2XCLK}}$. Accordingly the CLK2 signal of waveform (e) transitions to the value of CLK1 on the falling edge of the 2XCLK. As indicated CLK2 has a 90° phase displacement from CLK1.

The synchronization between waveforms (b)-(e) is maintained even when the PLL has not locked. Although a 180° phase difference is illustrated

20   with respect to REFCLK and CLK1, the PLL will adjust to bring REFCLK and CLK1 into phase alignment and the PLL will "lock" to maintain the alignment between CLK1 and REFCLK. Accordingly, the phase difference illustrated between CLK1 and REFCLK is only an initial condition. The synchronization between waveforms (a) and the remaining waveforms (b)-(e) will vary until

25   the PLL locks at which point waveform (d) will be synchronized and in phase with waveform (a).

Figure 3 illustrates one embodiment of creating multiple instances of the quadrature clock. The quadrature clock distribution of Figure 3 incorporates the benefits of both central generation and distributed

30   generation through the use of multiple PLL/DLLs without the aforementioned disadvantages of either architecture.

6

The power requirements associated with central generation of the quadrature are avoided by using localized flip-flops to distribute the derived quadrature clock to a local region. Although the 1XALIGN signal is distributed to all local regions, the 1XALIGN signal is used for alignment only and is not used to drive either CLK1 or CLK2. The 2XCLK signal need only clock a few components within each local region rather than all clocked components throughout the integrated circuit.

Unlike the distributed PLL/DLL architecture, a single alignment signal enables selection of the proper edges of the 2XCLK signal associated with each of the CLK1 and CLK2 signals. Moreover, considerable skewing or drifting of 1XALIGN is tolerable because the 1XALIGN is used to identify which edges of the 2XCLK signal to select.

Given that CLK2 is derived from CLK1 rather than independently derived from the 1XALIGN, the 1XALIGN signal can shift almost 90° in either direction while associating the same edges of the 2XCLK with CLK1 or CLK2. Thus distribution of the 1XALIGN is substantially immune from ordinary clock skew considerations.

The 1XALIGN 376 is distributed to each local region associated with a recovery circuit. The double clock 316 (2XCLK) is similarly distributed to each local region. In the illustrated embodiment, the double clock is distributed as a differential double clock that is converted back to a single ended double clock at each local region. In particular, single-sided-to-differential converter 320 converts the single sided 2XCLK to a differential 2XCLK signal that is provided to distribution block 322. Proximate the flip-flops, the differential-to-single-sided converter 330, 332 associated with the local region converts the differential 2XCLK back to a single-sided 2XCLK.

Each local region includes a recovery circuitry 398B for recovering CLK1 and CLK2 from the distributed 2XCLK and 1XALIGN signals. In the illustrated embodiment, recovery circuit 398B includes flip-flops 350B, 360B, and inverter 340B for recovering or reproducing the CLK1 380B and CLK2 390B clocks. In one embodiment, an integrated circuit has a plurality of

local regions requiring local generation of the quadrature clock from the distributed double clock and alignment signals.

CLK1 380A and CLK1 380B are distinctly generated instances of the same clock signal, CLK1. Similarly, CLK2 390A and CLK2 390B are distinctly generated instances of the same clock signal, CLK2. Thus CLK1 380A = CLK1 380B even though they are independently derived from the 2XCLK and 1XALIGN signals. Similarly, CLK2 390A=CLK2 390B even though they are independently derived from the 2XCLK and 1XALIGN signals. PLL 310 only requires one instance of the first clock for feedback. Thus only one of CLK1 380A...380B is provided to PLL 310 for feedback.

Figure 4 illustrates one embodiment of a method for generating a quadrature clock. In step 410, a double clock having a frequency twice that of a received reference clock is generated. An alignment signal corresponding to an inverted double clock divided by two is generated in step 420. A first clock is derived from the double clock in accordance with the alignment signal in step 430. The alignment signal identifies which edges of the double clock are associated with the first clock. The second clock is also derived from the alignment signal. In Figures 1, 3 the first clock is latched to provide the second clock in accordance with the inverted double clock. Thus in step 440, the second clock is derived from the first clock. The first and second clocks have substantially a 90° phase difference.

The ability to halt and step the quadrature clock is useful for testing the integrated circuit incorporating the quadrature clock of Figures 1 and 3. The ability to stop the quadrature clock at a particular level and subsequently step the quadrature clock requires additional circuitry.

Figure 5 illustrates one embodiment of a quadrature clock generator including circuitry for halting and stepping the quadrature clock. Controlling the quadrature clock in such a fashion may be provided in a test mode of operation in contrast with a normal mode of operation. PLL 510 cannot simply be stopped and re-started in a locked condition. Logic 530, 560, and 550 effectively serve as a "clutch" mechanism to couple or de-couple the divider and recovery circuits between the PLL 510 and a gated clock 520 when

halting and stepping the quadrature clock from a particular level or phase is desired in a test mode.

Halt control 540 generates the halt multiplexer control 546 used to control selection of the double clock and selection of the PLL feedback path.

5   Multiplexer 530 selects the double clock 516 (2XCLK) as either the PLL output 514 or the gated clock 520 in accordance with the halt multiplexer control 546. The gated clock 520 gates the PLL output 514 in accordance with a step signal 522.

When multiplexer 530 selects the gated clock 520 to provide the double

10   clock 516, the CLK1 580 signal is unsuitable as a feedback signal to the PLL 510 given that PLL 510 is no longer directly driving the double clock that CLK1 is derived from. Accordingly, multiplexer 550 and divider 560 provide a feedback clock signal having a frequency half that of the PLL output 514 in lieu of CLK1 580 to the PLL. This alternate feedback path may be referred to

15   as the bypass feedback path or test feedback path.

The gated clock 520 ensures that any changes in double clock are synchronized with the PLL output 514 in the test mode. The halt control 546 ensures that transitions between the halt and "free-running" conditions during a test mode are synchronized with transitions in the PLL output state

20   514.

Figure 5 thus illustrates a clock generator providing a clock output having a frequency that is twice that of a received reference clock. A multiplexer is coupled to select one of the clock output and a gated clock output as a double clock in accordance with a halt multiplexer control.

25   Divider circuitry divides the double clock to provide an alignment signal with a frequency half that of the double clock. A recovery circuit recovers a first clock and a second clock from the double clock in accordance with the alignment signal, wherein the first and second clocks have substantially a 90° phase difference.

30   The halt control should be disabled during normal operation. This is accomplished in one embodiment by tying HALT 542 to a logic level that

ensures halt multiplexer control 546 always selects PLL output 514 during normal operational modes.

Multiplexer 550 provides a selected one of a normal feedback path or a test feedback path to PLL 510 in accordance with the signal, TEST 552. During normal operation, the normal feedback path with CLK1 580 is selected as the feedback clock signal for the PLL. When testing, however, CLK1 580 is not being directly driven by the PLL and is not a suitable feedback signal. Accordingly, a divided PLL output (due to divider 560) is selected as the feedback clock signal from a test feedback path when testing. TEST 552 is tied to a logic level that ensures multiplexer 550 selects CLK1 580 as the feedback clock for PLL 510 during the normal operational modes. TEST 552 provides the test feedback path to PLL during test modes.

Figure 6 illustrates one embodiment of the halt control 540 of Figure 5. The halt control circuitry generates halt multiplexer control 646 from inputs stop level 644, 1XALIGN 676, halt 642, and PLL out 614.

The 1XALIGN 676 and STOP LEVEL 644 are provided as inputs to exclusive NOR gate 692. STOP LEVEL 644 indicates whether CLK1 should be stopped at either a high or a low level. NAND gate 694 receives a halt signal 642 and the output of exclusive NOR gate 692. The output of exclusive NOR gate 692 is provided as the data input to flip-flop 690. In one embodiment flip-flop 690 is a D-type flip-flop.

Cross-coupled NAND gates 696 and 698 form a latch providing the halt multiplexer control 646. Tying HALT 642 to a low logic level ensures that the halt multiplexer control is also a low logic level. Referring to Figure 5, the halt control circuit 540 should be disabled during normal operation of the quadrature clock generator when the PLL output 514 is the double clock. Multiplexer 530 selects the gated clock 520 when A=halt multiplexer control 546 or the PLL output when B=halt multiplexer control. For the halt control circuit of Figure 6, this implies that B=0 and A=1 because the halt multiplexer control is 0 when disabled.

When HALT is enabled, the halt multiplexer control selects the gated clock as the double clock when the 1XALIGN 676 and STOP LEVEL 644

match. STOP LEVEL indicates which clock level to stop at (i.e., low, high). Note the use of 1XALIGN rather than the CLK1 signal in the combinatorial logic of Figure 6. Due to the high frequency of the 2XCLK and the electrical length of the path to the multiple instances of the recovery circuits, the

5   1XALIGN rather than the CLK1 signal should be used.

Figure 7 illustrates one embodiment of a method of generating and controlling the quadrature clock. In step 710, one of a generated clock (e.g., PLL out 514) and a gated generated clock (e.g., gated clock 520) is selected as a double clock in accordance with a halt multiplexer control. The double clock

10   has a frequency twice that of a received reference clock.

An alignment signal corresponding to an inverted double clock divided by two is generated in step 720. A plurality of instances of a first clock and associated second clock are derived from the double clock in accordance with the alignment signal in step 730. Each first and associated second clock

15   instance have a relative 90° phase difference.

In step 740, the halt multiplexer control is controlled to select the gated generated clock signal when the alignment signal matches a pre-determined clock level. The halt multiplexer control is clocked by the generated clock for synchronization with the generated clock.

20   The halt multiplexer control thus permits the user to stop all instances of the quadrature clock at a particular level in synchronization with the generated clock. Once the gated generated clock is selected, the user may step the quadrature clock instances through the quadrature clock cycles such that the quadrature clock instances are synchronized with each other.

25   Figure 8 illustrates one embodiment of quadrature clock generation and distribution across an integrated circuit die. A clock generator is fabricated on a given area 850 of an integrated circuit die 830. The 1XALIGN and the double clock (2XCLK) are distributed to each of a plurality of local regions 840A, 840B, 840C, …840D. Each local region has a recovery circuit for

30   deriving a local instance of CLK1 and CLK2 from the 2XCLK and 1XALIGN signals.

11

The CLK1 instances are relatively phase aligned such that CLK1A = CLK1B = CLK1C...=CLK1D). Similarly, the CLK2 instances are relatively phase aligned such that CLK2A=CLK2B=CLK2C...=CLK2C). Each CLK2 instance has a 90° phase difference from its associated CLK1 instance (as well

5   as the other instances of CLK1).

The integrated circuit die 830 resides within an integrated circuit package 820. A REFCLK is provided to the clock generator 830 via an external pin 822 of the integrated circuit package 820.

Various modifications and changes may be made thereto without

10   departing from the broader spirit and scope of the invention as set forth in the claims. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.